



UNIVERSITY OF OXFORD
MATHEMATICAL INSTITUTE

Random Fractals:
[subtitle]

Author
Paul DUBOIS

Supervisor
Ben HAMBLY

April 26, 2021

Random Fractals

Abstract

[blah]

Acknowledgment

First of all, I would like to thank my supervisor, Ben Hambly, for guiding and supporting through this project.

I would like to thank the whole administration team of Oxford, for all the help they provided despite the pandemic context.

Contents

0	Introduction	4
1	Background Theory	5
1.1	Dimensions	5
1.1.1	Intuition	5
1.1.2	Topological Dimension	6
1.1.3	Box Dimension	6
1.1.4	Hausdorff Dimension	6
1.1.5	Some Relations Between Dimensions	7
1.2	Fractals	8
1.2.1	Formal Definition	8
1.2.2	Famous Examples	8
1.2.3	Real Life Fractals	11
2	Percolation Fractals	12
2.1	Classical Percolation	12
2.2	Recursive Percolation	12
2.3	Extension to other dimensions	13
2.4	Density	14
2.5	Dimensionality	14
2.6	Blob	16
2.6.1	Manhattan (step) distance	16
2.6.2	Euclidean distance	16
2.6.3	Area	16
2.6.4	Volume	16
2.7	Random Walks	16
3	Percolation Crossings	17
3.1	Types of Crossings	17
3.2	Finding Crossings	17
3.3	Crossings Probability	21
3.4	Crossings Length	21
3.5	Crossings Dimension	21
4	Projections	22
4.1	2D to 1D	22
4.2	3D to 2D	22
4.3	3D to 1D	22
4.4	Dimension of the (non void) Universe	22
A	More on Fractals	1
A.1	Mandelbrot and Julia Sets	1
A.2	Sierpiski Fractals	1
A.3	Coastline of Islands	2
B	Plots	4

0 Introduction

1 Background Theory

We derive here a mathematical basis on fractals and dimensions of sets. This will be used later in our more specific study of random fractals.

1.1 Dimensions

The concept of dimension is quite intuitive from an everyday life perspective. However, the mathematical concept is more involved. From the non-mathematical world, this can be used to have better understanding of other objects arising from other fields such as DNA structure¹, or lungs[Ionescu et al.(2009)Ionescu, Oust

1.1.1 Intuition

Some objects that we are used to working with have an established dimension:

- **Empty set / Point**: dimension 0
- **Curve** (e.g.: *line*): dimension 1
- **Surface** (e.g.: *square*): dimension 2
- **Volume** (e.g.: *cube*): dimension 3

All of these usual objects have integral dimensions, making it (relatively) easy to understand.

A rule of thumb to calculate the dimension is to double (or, in general, multiply by n) the size of the object, and count the number of copies of the original object obtained. If there are N original objects, the dimension is $d = \frac{\ln(N)}{\ln(n)}$. This is so that when scaling by n , the length/area/volume of the set is multiplied by $N = n^d$.

Some objects have a more complicated dimension (in fact, a non-integral one)²:

- **Cantor Set** (see fig. 3): dimension $\log_3(2) = \frac{\ln(2)}{\ln(3)} \simeq 0.631$
- **Koch Snowflake** (see fig. 4): dimension $\log_3(4) = \frac{\ln(4)}{\ln(3)} \simeq 1.262$
- **Sierpinski Carpet** (see fig. 5): dimension $\log_2(8) = \frac{\ln(8)}{\ln(3)} \simeq 1.893$

The fractional dimensions of these objects justify creating a formal mathematical definition.

After this quick overview, 3 properties seem desirable for a definition of dimension [Pol(2009)]. For a set X ($\subset \mathbb{R}^n$, in general):

1. If X is a manifold, then the dimension should coincide with our natural preconception.
2. In some cases, X may have a fractional (i.e. non-integral) dimension. .
This is needed in order to describe sets as the ones discussed above.
3. If X is countable, then X has dimension 0.

There are several definitions for dimension, satisfying different properties.

¹<https://news.mit.edu/2009/3d-genome>

²This will be discussed in more detail in 1.2.2.

1.1.2 Topological Dimension

The topological dimension is the most straightforward way to define dimension. It relies on the intuition that the boundary of a ball of dimension d should have dimension $d - 1$.

Definition 1.1 (Topological dimension). *The topological dimension $\dim_T(X)$ of a set X is defined recursively through the following:*

$$\dim_T(X) = \begin{cases} -1 & \text{if } X = \emptyset \\ d & \text{if } d = \min \{n \in \mathbb{N} \mid \forall x \in X, \exists r > 0 \text{ s.t. } \dim_T(\partial B_r(x) \cap X) \leq n - 1\} \end{cases}$$

This definition satisfies the first desired property (1.1.1:1). However, \dim_T is always an integer (this is clear from definition). Therefore, it does not satisfy the second desired property (1.1.1:2).

1.1.3 Box Dimension

The box dimension is more general than the topological one, in the sense that it allows non-integral dimensions. It relies on an intuition mentioned before: when scaled by n , a set contains N copies of the original set, then the dimension should be $\frac{\ln(N)}{\ln(n)}$.

Definition 1.2 (Box dimension). *The box dimension \dim_B of a set X is defined through the following limit:*

$$\dim_B(X) = \lim_{\varepsilon \rightarrow 0} \frac{\ln(N(\varepsilon))}{-\log(\varepsilon)}$$

Here, $N(\varepsilon)$ is the smallest number of ε -balls needed to cover X .

Note that box dimension exists only if this limit exists. In the case this limit does not exist, it is possible to define upper and lower box dimensions, taking respectively \limsup and \liminf in the definition above. (Both upper and lower box dimension with the general box dimension when it exists.)

This definition satisfies the first and second desired property (1.1.1:1,2). However, if we consider $X = \{0\} \cup \{\frac{1}{n} \mid n \in \mathbb{N}^*\}$, then $\dim_B(X) > 0$, but X is countable. Therefore, it does not satisfies the third desired property (1.1.1:3).

1.1.4 Hausdorff Dimension

The Hausdorff dimension (sometimes also called fractal dimension) is considered to be the most robust concept of dimension. The reason of this consortium is that Hausdorff dimension has a measure (the Hausdorff measure) associated to it). The definition is more involved than both the topological and the box dimensions.

Definition 1.3 (Hausdorff/fractal dimension). *The Hausdorff dimension \dim_H of a set X is defined as follows:*

$$\text{For } \varepsilon > 0, d \geq 0, H_\varepsilon^d(X) = \inf_{\substack{\mathcal{U} \text{ open cover of } X \\ U \in \mathcal{U} \implies \text{diam}(U) < \varepsilon}} \left\{ \sum_{U \in \mathcal{U}} \text{diam}(U)^d \right\}$$

A $H_\varepsilon^d(X)$ is an increasing function as $\varepsilon \rightarrow 0$, the following limit is well defined:

$$H^d(X) = \lim_{\varepsilon \rightarrow 0} H_\varepsilon^d(X)$$

This defines a measure, which is called the d -dimensional Hausdorff measure.

For a set X , $H^d(X)$ jumps from 0 to ∞ , when d varies from 0 to ∞ (this will be proved as a claim, see 1.1.5). The particular value at which this jump occurs is the Hausdorff dimension of X . Formally:

$$\dim_H(X) = \inf \{\delta \mid H^\delta(X) = 0\}$$

This definition satisfies all three of the desired property (1.1.1:1,2,3). The first two are clear from definition.

Property 1.1. *Countable sets have Hausdorff dimension 0.*

Proof. Suppose $X = \{x_n \mid n \in \mathbb{N}\}$ is countable. Let $\varepsilon > 0$, and take $\{\varepsilon_n \mid n \in \mathbb{N}\}$ such that $\sum_{n=0}^{\infty} \varepsilon_n^d < \varepsilon$. Then $\mathcal{B} = \{B(x_n, \varepsilon_n) \mid n \in \mathbb{N}\}$ is an open cover of X , so $H_\varepsilon^d(X) \leq \varepsilon$. As this is true $\forall \varepsilon > 0$, $\forall d \geq 0$, get $H^d(X) = 0$ for all $d \geq 0$. So finally, $\dim_H(X) = 0$. \square

The Hausdorff dimension is usually harder to calculate in practice.

1.1.5 Some Relations Between Dimensions

For dimension definition, there is a choice to make between more robust, but hard to calculate (Hausdorff dimension) and less robust but easier to calculate (Topological/Box dimension) definitions. It is therefore very useful to know some relationships between the three notions (it is then possible to use, for example, box dimension to give estimates for the Hausdorff dimension).

Property 1.2 (Upper bound for fractal dimension). *Box dimension is greater than or equal to Hausdorff dimension.*

Proof. For a set X (with box dimension well defined): Let $\eta > 0$, $\gamma = \dim_B(X) + \eta$ and $\delta = \dim_B(X) + 2\eta$. Then, $\exists \varepsilon > 0$ such that X can be covered by $N(\varepsilon) < \varepsilon^{-\gamma}$ ε -balls. Thus, $H_\varepsilon^\delta(X) \leq \varepsilon^{-\gamma} \varepsilon^\delta = \varepsilon^\eta$, so $H^\delta(X) = 0$. This gives $\dim_H(X) < \dim_B(X) + 2\eta \quad \forall \eta > 0$, and hence, $\dim_H(X) \leq \dim_B(X)$. \square

Thus, by calculating the box dimension, we also have an upper bound for the Hausdorff dimension.

Lemma 1.1. *If the d -dimensional Lebesgue measure is non-zero, then the Hausdorff dimension is greater than or equal to d .*

Proof. Suppose a set X is such that $\dim_H(X) < d$.

Claim. $H^d(X) < \infty \implies H^c(X) = 0 \quad \forall c > d$

Proof. As $H^d(X) < \infty$: For all $\varepsilon > 0$ there is an open cover \mathcal{U} for X such that $\sum_{U \in \mathcal{U}} \text{diam}(U)^d < \infty$ and $\text{diam}(U) < \varepsilon$. So

$$\sum_{U \in \mathcal{U}} \text{diam}(U)^c \leq \underbrace{\varepsilon^{c-d}}_{\substack{\rightarrow 0 \\ \text{as } \varepsilon \rightarrow 0}} \underbrace{\sum_{U \in \mathcal{U}} \text{diam}(U)^d}_{< \infty} \rightarrow 0 \quad \text{as } \varepsilon \rightarrow 0$$

\square

Thus, $H^d(X) = 0$. Now, the d -dimensional Hausdorff measure is a rescaling of the usual d -dimensional Lebesgue measure, so $\Lambda_d(X) = 0$ ³. This completes the proof by taking the contrapositive. \square

Thus, finding a d such that the Lebesgue measure is non-zero gives a lower bound for the Hausdorff dimension.

Property 1.3 (Lower bound for fractal dimension). *Topological dimension is less than or equal to Hausdorff dimension.*

³Writing $\Lambda_d(X)$ for the d -dimensional Lebesgue measure of X .

Proof. This follows directly from the last property, as a set X having Topological dimension d (i.e. $\dim_T(X) = d$) implies that its d -dimensional Lebesgue measure will be positive (i.e. $\Lambda_d(X) > 0$). \square

In fact, the Hausdorff dimension is bounded by the topological dimension and the box dimension, i.e. for any set X , $\dim_T(X) \leq \dim_H(X) \leq \dim_B(X)$.

1.2 Fractals

The first traces of fractals come back to the 17th century with the mathematician and philosopher Gottfried Leibniz, philosophizing about recursive self-similarity. Proper sketches of a mathematical definition for fractals only go back to Karl Weierstrass, during the 19th century. The recursive self-similarity pattern of most common fractals made their fame: images of some fractals (e.g. Julia sets, Koch snowflake, Sierpiski gasket) have become popular across the mathematics and non-mathematics world.

1.2.1 Formal Definition

There is a controversial mathematical definition of fractals (introduced by Benoît Mandelbrot):

Definition 1.4 (Fractal). *A fractal is a subset of Euclidean space with a Hausdorff dimension that strictly exceeds its topological dimension (i.e. X is a fractal if $\dim_T(X) < \dim_H(X)$).*

However, since sets with "recursive patterns" are often fractals. The "self-similarity" property is sometimes itself taken as the definition for fractals.

1.2.2 Famous Examples

Some fractals became very famous, both for their aesthetic appeal and as an example of a complex structure arising from simple rules. Fractals are one of the best-known examples of mathematical visualization and mathematical beauty.

Basin Boundaries of Complex Maps Along the most famous fractals, one finds the ones that arise from looking at the contour of sets (basin boundaries) derived from complex maps.

Contour of the Mandelbrot Set The Mandelbrot set is a subset of the complex plane defined as follows:

Definition 1.5 (Mandelbrot Set). *Let $z_0 = 0$ and $z_{n+1} = f_c(z_n)$, with $f_c(z) = z^2 + c$. Then, the Mandelbrot set $M \subseteq \mathbb{C}$ is $M = \{c \in \mathbb{C} \mid z_n \text{ does not diverge}\}$.*

The Mandelbrot set is not a fractal itself (it has dimension 2, as it contains $\{w \in \mathbb{C} \mid |w - 1| < \frac{1}{4}\}$ [3DX(2020)], and is contained in \mathbb{C} , both having dimension 2).

It is surprising that the boundary ∂M of M also has Hausdorff dimension 2. This was proved by Shishikura in 1992 [Shishikura(1992)], as a consequence of ∂M

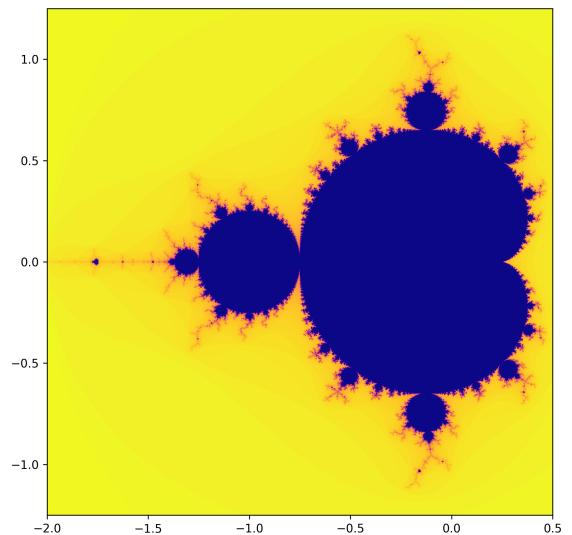


Figure 1: Mandelbrot Set Plot⁴

⁴More details on the plots in the appendix.

having a positive 2-dimensional Lebesgue measure (i.e. $\Lambda_2(\partial M) > 0$).

Despite having an integral fractional dimension, ∂M is commonly considered to be a fractal (because the integrality of fractional dimension is not obvious, and the self-similarity of the set).

Contour of Julia Sets Julia sets are also subset of the complex plane defined similarly to the Mandelbrot set:

Definition 1.6 (Julia Set). *For a complex $c \in \mathbb{C}$ constant: Let $z_0 = z$ and $z_{n+1} = f_c(z_n)$, $f_c(z) = z^2 + c$ as before. The filled Julia set about c is $K(f_c) = \{z \in \mathbb{C} \mid z_n \text{ does not diverges}\} \subseteq \mathbb{C}$. The Julia set (about c) $J(f_c)$ is the boundary of $K(f_c)$ ($J(f_c) \subseteq \mathbb{C}$).*

The dimension of $J(f_c)$ will of course depend on c . It is considered as a fractal even when the dimension is an integer. For some values of c , dimension of $J(f_c)$ is well known:

$c = 0$	$\dim_H(J(f_c)) = 1$	"Circle"
$c = \frac{1}{4}$	$\dim_H(J(f_c)) \approx 1.0812$	
$c = i$	$\dim_H(J(f_c)) \approx 1.2$	"Dendrite"
$c = -1$	$\dim_H(J(f_c)) \approx 1.2683$	
$c = -0.123 + 0.745i$	$\dim_H(J(f_c)) \approx 1.3934$	"Douady rabbit"

The Mandelbrot set and Julia sets are very closely related. In fact, Shishikura proved [Shishikura(1992)] that when c is on ∂M , then the Julia set $J(f_c)$ associated to it has (Hausdorff) dimension 2. Moreover, Heinemann and Stratmann have shown [Heinemann and Stratmann(1998)] that when the quadratic f_c is parametrized with c near the boundary of the Mandelbrot set M , the Hausdorff dimension of $J(f_c)$ is arbitrarily close to 2.

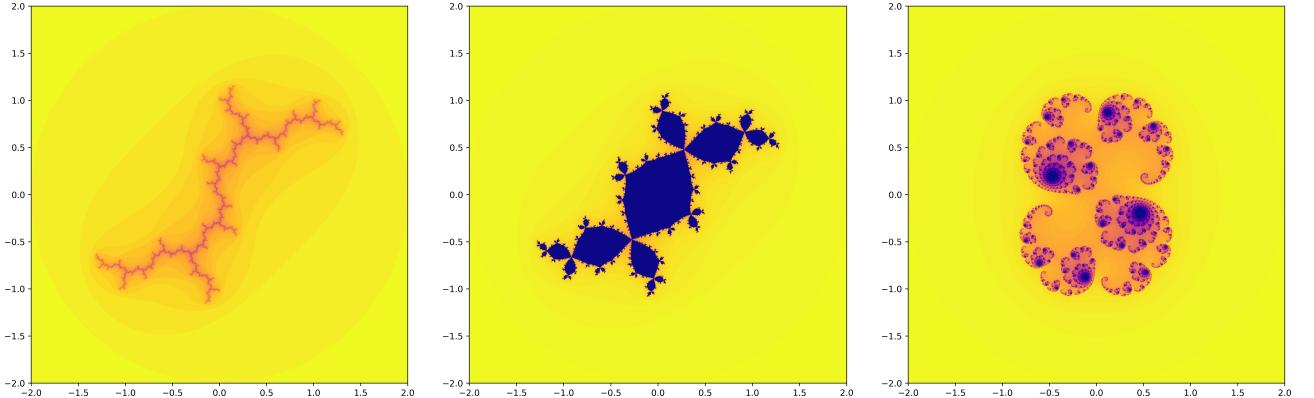


Figure 2: Popular Julia Set Plots
(left to right: "Dendrite"⁵; "Douady rabbit"⁶; other typical Julia set⁷)

Cantor Set The (middle third) Cantor set $C \subseteq [0, 1]$ is constructed via the following informal definition:

- Start with $C_0 = [0, 1]$
- Generate iteratively $C_n = \frac{C_{n-1}}{3} + \frac{2+C_{n-1}}{3}$
- Take the limit $C = \lim_{n \rightarrow \infty} C_n$



Figure 3: Cantor Set (first 6 iterations)

⁵ $c = i$

⁶ $c = -0.123 + 0.745i$

⁷ $c = 0.285 + 0.01i$

The Cantor set satisfies $C = \frac{C}{3} + \frac{2+C}{3}$, so rescaling by $\frac{1}{3}$, it contains two copies of itself. So the intuitive Hausdorff dimension for the Cantor set is $\dim_H(C) = \frac{\log(2)}{\log(3)} \approx 0.631$. This can be proved rigorously (see [Falconer(1990), p. 34-35, ex. 2.7].

Another equivalent definition for the cantor set C is "points in $[0, 1]$ with extension in base 3 composed of 0 and 2 only". That is:

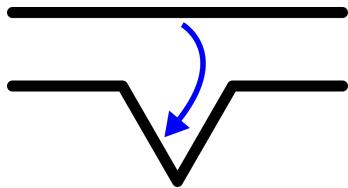
$$C = \left\{ x \in [0, 1] \mid x = \sum_{k=1}^{\infty} x_k 3^{-k}, \quad x_k \in \{0, 2\} \quad \forall k \right\}$$

From this definition, it is straightforward that the Cantor set is totally disconnected. In fact, fractals with Hausdorff dimension smaller than 1 are totally disconnected ([Falconer(1990), p. 33, prop. 2.5]).

Koch Snowflake The Koch Snowflake is an example of a fractal curve. Fractal curves are obtained by applying the same transformation recursively to each segment of the previous iteration of the curve. Their fractional dimension is in the interval $[1, 2]$ (or in $[1, 3]$ if the curve evolves in a three dimensional space).

The Koch snowflake is obtained after replacing each edge of an equilateral triangle by a Koch curve with the triangle edge as initial segment.

Now, the Koch curve is defined recursively as follows: K_n is the curve at iteration n , with segments K_n^1, \dots, K_n^m . For each $i \in \llbracket 1, m \rrbracket$, split K_n^i into 3 equal length segments, and replace the middle one with two new ones as on this diagram:



Merging back all segment gives K_{n+1} .

The Koch curve starting form a segment $[a, b]$ is the limit curve $K = \lim_{n \rightarrow \infty} K_n$ with $K_0 = [a, b]$.

At each level, the Koch curve is scaled by $1/3$, and 4 copies are created. Thus, the intuitive Hausdorff dimension for the Koch curve (which is also the dimension of the Koch snowflake) is $\dim_H(K) = \frac{\log(4)}{\log(3)} \approx 1.262$. This can be proved rigorously using similar techniques as in [Falconer(1990), p. 34-35, ex. 2.7].

Sierpiski Carpet Sierpiski carpet is a fractal constructed recursively by removing parts of its initial set.

The construction starts from a (filled) square. Then, at each step, split every square into 9 sub-square as follows:

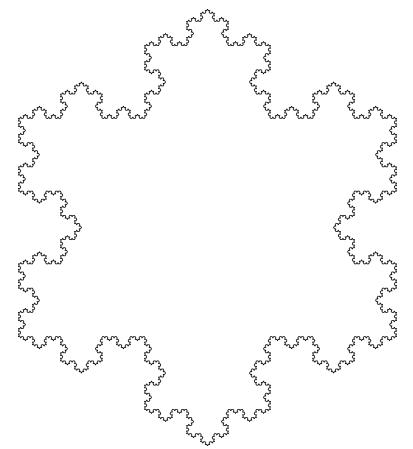
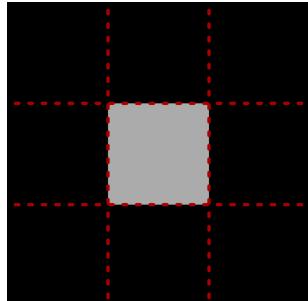


Figure 4: Koch Snowflake Curve Plot (5 iterations)

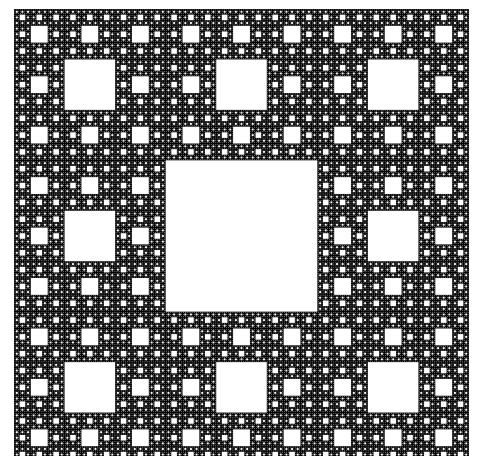


Figure 5: Sierpinski Carpet (6 steps)

Remove the central square, and repeat the operation on the remaining 8 squares.

Note that this can be seen as a version of the Cantor set starting in 2 dimensions.

The figure is made of 8 copies of itself, scaled by a factor of $1/3$. Therefore, the intuitive Hausdorff dimension for this set is $\dim_H(K) = \frac{\log(8)}{\log(3)} \approx 1.893$. Again, this can be proved rigorously using similar techniques as in [Falconer(1990), p. 34-35, ex. 2.7].

1.2.3 Real Life Fractals

Fractals are more than mathematical constructions, in fact self-similarity structures also occur in nature. The limit is never reached, because of the physical constraints, but we can view them as natural approximations of fractals. This gives a good motivation for studying fractals in a mathematical context.

Plants Some plants (not even genetically modified) have self-similarity patterns, and may therefore be considered as fractals.

Ferns Perhaps the most obvious and popular natural fractal: Ferns leaves. The leaves are self-similar, with varying pattern across variety of ferns.

Cauliflower A less obvious example: Cauliflowers' surface is a fractal. In fact, since each branch splits into about 13 branches, each about 3 times shorter, the approximate dimension of a Cauliflower's surface is $\log_3(13) \approx 2.335$.



(a) Fern Leave



(b) Cauliflower

Figure 6: Plant Fractals

Coastline of Islands When trying to measure the length of coastlines, scientists realized that the more precise their attempt was, the higher was the value of the length found. In fact, this makes sense: the approximation of a curve with shorter segments will capture more of the smaller curve details, resulting in a longer length. The approximate length found could be made really large after approximating using smaller and smaller segments (until physical boundary are reached). This yields the fact that the coastline is a fractal with dimension greater than one.

Using electronic maps, we try to estimate the fractal dimension of the coastlines for the UK islands, Iceland and Madagascar. More details on the calculations techniques used are given in the appendix (see A.3). The (approximate) coastline dimension for UK, Iceland, and Madagascar are respectively 1.24, 1.25, and 1.06.

The coastlines of the UK and Iceland are much more irregular than the one of Madagascar. The approximate values found for coastlines dimensions therefore make sense.

2 Percolation Fractals

Percolated fractals will be the main object we intend to study. Starting from a unit square, and remove some material to obtain a fractal.

We will begin with the definition of the percolation process in the two dimensional case, as it will be our main interest (it is also the most intuitive case, as easy to picture). We will then extend the definition to other dimensions.

There are two types of percolations: the "classical", and the "recursive", both have their interest, and we will compare the two throughout this study.

2.1 Classical Percolation

The "classical" percolation is the simplest one, as it is only composed of one filtration. The formal definition goes as follows: Split the unit square $[0, 1]^2$ into n^2 squares of side $1/n$, indexed by $1 \leq i, j \leq n$:

$$B_{i,j} = \left[\frac{i-1}{n}, \frac{i}{n} \right] \times \left[\frac{j-1}{n}, \frac{j}{n} \right].$$

Figure 7: Classical Percolation
($n = 4, p = 0.6$)

Then, each squares will be selected or thrown away, according to random variables $\epsilon_{i,j}$ following a Bernoulli distribution with probability parameter p ⁸, $\epsilon_{i,j} \sim \mathcal{B}(p)$. Finally, the classical percolation P is

$$P = \bigcup_{i,j \text{ s.t. } \epsilon_{i,j}=1} B_{i,j}.$$

We will adopt the notation $P \sim \text{Perc}(n, p, 1)$ for such a setup (a classical percolation of side n and probability parameter p).

In this case, it is interesting to study the behaviour as $n \rightarrow \infty$, we will write $P \sim \text{Perc}(\infty, p, 1)$ for $P = \lim_{n \rightarrow \infty} P^n$, $P^n \sim \text{Perc}(n, p, 1)$.

2.2 Recursive Percolation

The "recursive" percolation is a little more involved. Beginning with a classical percolation, we apply another one to each of the remaining squares, and continue recursively.

Formally, for each $1 \leq k \leq d$, split the unit square into $(n^k)^D$ squares of side $1/n^k$ indexed by $1 \leq i, j \leq n^k$

$$B_{i,j}^k = \left[\frac{i-1}{n^k}, \frac{i}{n^k} \right] \times \left[\frac{j-1}{n^k}, \frac{j}{n^k} \right].$$

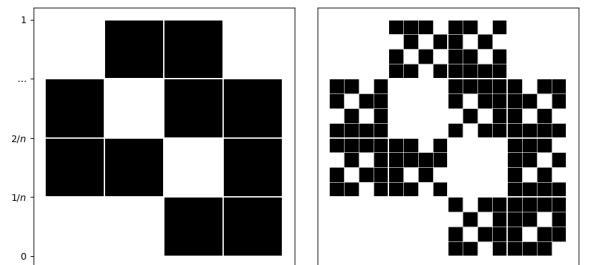


Figure 8: Recursive Percolation
($n = 4, p = 0.6, k = 1, 2$)

Again, associate to each square a Bernoulli random variable with parameter p , $\epsilon_{i,j}^k \sim \mathcal{B}(p)$. Finally, the recursive percolation P^d is defined recursively by:

$$P^k = P^{k-1} \cap \left(\bigcup_{i,j \text{ s.t. } \epsilon_{i,j}^k=1} B_{i,j}^k \right) \quad \forall 1 \leq k \leq d$$

and

$$P^0 = [0, 1]^2.$$

⁸Writing $x \sim \mathcal{B}(p)$ for x following a Bernoulli distribution with parameter p , i.e. $\mathbb{P}(x = 1) = p$ and $\mathbb{P}(x = 0) = 1 - p$.

We will adopt the notation $P^d \sim \text{Perc}(n, p, d)$ for such a setup (a recursive percolation of depth d , side n and probability parameter p).

In this case, it is interesting to study the behaviour as $d \rightarrow \infty$, with n fixed. We will write $P \sim \text{Perc}(n, p)$ for $P = \bigcap_{k \rightarrow \infty} P^k$, with P^k defined recursively as above.

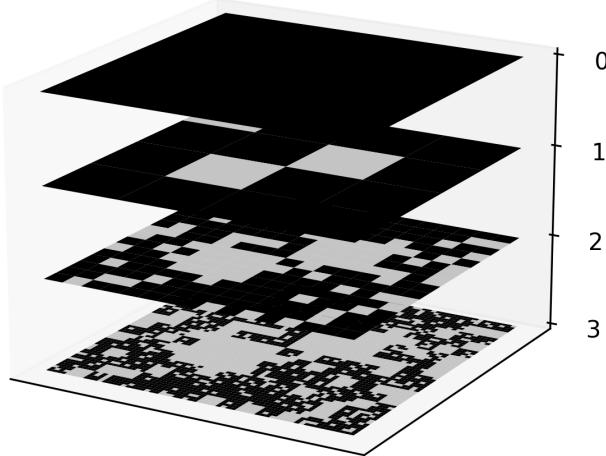


Figure 9: Recursive Percolation
($n = 4, d = 3, p = 0.8$)

2.3 Extension to other dimensions

Now that we defined percolations in two dimensions, it is straightforward to extend the definition to other dimensions.

Extension to D dimensions It suffices to add parameters in the definition to extend it to other dimension, formally:

Classical Split the unit cuboid $[0, 1]^D$ into n^D squares of side $1/n$, indexed by $1 \leq i_1, \dots, i_D \leq n$:

$$B_{i_1, \dots, i_D} = \left[\frac{i_1 - 1}{n}, \frac{i_1}{n} \right] \times \cdots \times \left[\frac{i_D - 1}{n}, \frac{i_D}{n} \right].$$

Then, attach to each cuboid a Bernoulli random variable $\epsilon_{i_1, \dots, i_D} \sim \mathcal{B}(p)$. The classical percolation P in D dimension is

$$P = \bigcup_{i_1, \dots, i_D \text{ s.t. } \epsilon_{i_1, \dots, i_D} = 1} B_{i_1, \dots, i_D}.$$

We will denote this setup by $P \sim \text{Perc}^D(n, p, 1)$ (a classical percolation of side n and probability parameter p in D dimensions), and write $P \sim \text{Perc}^D(\infty, p, 1)$ for $P = \lim_{n \rightarrow \infty} P^n$, $P^n \sim \text{Perc}^D(n, p, 1)$.

Recursive For each $1 \leq k \leq d$, split the unit cuboid into $(n^k)^D$ cuboid of side $1/n^k$ indexed by $1 \leq i_1, \dots, i_D \leq n^k$

$$B_{i_1, \dots, i_D}^k = \left[\frac{i_1 - 1}{n^k}, \frac{i_1}{n^k} \right] \times \cdots \times \left[\frac{i_D - 1}{n^k}, \frac{i_D}{n^k} \right].$$

Again, attach to each cuboid a Bernoulli random variable $\epsilon_{i_1, \dots, i_D}^k \sim \mathcal{B}(p)$. Finally, the recursive percolation P^d is defined recursively by:

$$P^k = P^{k-1} \bigcap \left(\bigcup_{i_1, \dots, i_D \text{ s.t. } \epsilon_{i_1, \dots, i_D}^k = 1} B_{i_1, \dots, i_D}^k \right) \quad \forall 1 \leq k \leq d$$

and

$$P^0 = [0, 1]^2.$$

We will denote this setup by $P^d \sim \text{Perc}^D(n, p, d)$ (a recursive percolation of depth d , side n and probability parameter p in D dimensions). And again, we will write $P \sim \text{Perc}^D(n, p, \infty)$ for $P = \bigcap_{k \rightarrow \infty} P^k$, with P^k as above.

In practice, we will never use more than three dimensions. First, our world is in three dimensions, so it makes sense to stop there. In addition to that, the curse of dimensionality stops us (calculations and notations are too heavy from the mathematical point of view, and computations are too expensive from a modelling perspective).

Restriction to 1 dimension The restriction of the percolation process to one dimension can be thought as a randomized and generalized version of the Cantor set. The Cantor set splits the interval in 3 equal parts, and keep the two extreme ones, while the general recursive percolation process splits the interval into n equal parts, and keep each interval with probability p .

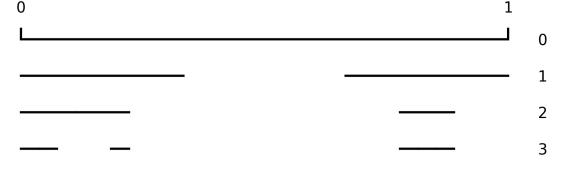


Figure 10: 1D Recursive Percolation ($n = 3, d = 4, p = 0.6$)

2.4 Density

Since the percolation process involves randomness, the density is not well defined. However, it is possible to calculate the expected density ⁹.

It is straightforward to remark that the expected density of a classical percolation $P \sim \text{Perc}^D(n, p, 1)$ is p (for all D and all n). Note that this density is constant as $n \rightarrow \infty$.

Now, for a recursive percolation P such that $P \sim \text{Perc}^D(n, p, d)$, the density is p^d . Note that this density tends to zero as d grows to infinity (except in the trivial case $p = 1$). Thus, if $P \sim \text{Perc}^D(n, p, \infty)$, the expected density is 0.

2.5 Dimensionality

Again, we can only give an expected dimension, as the process involves randomness. We ignore the cases $p = 0$ and $p = 1$ (these are trivial), and we concentrate on $p \in (0, 1)$.

We will treat separately the finite cases, and the two limit cases (limit as classical percolation, and as recursive percolation).

Finite Cases Take n and d finite, $P \sim \text{Perc}^D(n, p, d)$. Let Z be the number of remaining squares ($Z = |\{(i_1, \dots, i_D) \mid \epsilon_{i_1, \dots, i_D} = 1\}|$). The expected number of remaining squares is then $\mathbb{E}(Z) = p(n^d)^D$.

There are two possibilities for the dimension of P :

1. $Z = 0$: then $P = \emptyset$ and dimension of P is 0;
2. $Z > 0$: then $P \supseteq [\frac{i_1-1}{n^k}, \frac{i_1}{n^k}] \times \dots \times [\frac{i_D-1}{n^k}, \frac{i_D}{n^k}]$ for some (i_1, \dots, i_D) , so dimension of P is D .

Therefore, the expected dimension of P is:

$$\mathbb{E}(\dim_H(P)) = 0 \cdot \mathbb{P}(Z = 0) + D \cdot \mathbb{P}(Z > 0) = D \left(1 - (1-p)^{(n^d)^D} \right).$$

⁹The density of a set $X \subseteq [0, 1]^D$ is the proportion of points in the unit cuboid that are also in X .

In practice, this is close to D , as soon as n or d grows. Note that \dim_T would only differ by the fact that $\dim_T(\emptyset) = -1$ whereas $\dim_H(\emptyset) = 0$. Bow dimension \dim_B will be the same as the Hausdorff one.

Classical Percolation We are looking at $P \sim \text{Perc}^D(\infty, p, 1)$.

Heuristic Argument Intuitively, the expected dimension of this percolation should be D , since the density is positive.

Formal Calculations We show that the Hausdorff dimension is D almost surely.

Let \mathcal{U} be an open cover for P , and let V be an open subset of the unit cuboid of dimension D . Then, $\mathbb{P}(P \cap V = \emptyset) = 0$ (as V is uncountable). Thus, it is almost sure that \mathcal{U} covers $[0, 1]^D$ (otherwise, there is an open subset V of $[0, 1]^D$ not intersecting P). So $H_\epsilon^d(P) = H_\epsilon^d([0, 1]^D)$ almost surely. Therefore, $\dim_H(P) = \dim_H([0, 1]^D) = D$ almost surely.

Recursive Percolation We are interested in $P \sim \text{Perc}^D(n, p, \infty)$.

Eventually Empty Percolation In the case of $p < 1/n^D$, the number of expected cuboid is less than one. Thus, as $d \rightarrow \infty$, the percolation will almost surely become empty. Therefore, for $p < 1/n^D$, $\dim_H(P) = 0$ almost surely.

Next, we will suppose $p \geq 1/n^D$.

Heuristically In the case of a recursive percolation, P has self-similar properties that can help finding the expected dimension. After the first percolation, each selected cuboid is another version of P scaled by $1/n$. This goes on recursively. Therefore, the intuitive dimension to expect is

$$\mathbb{E}(\dim_H(P)) = \frac{\ln(\mathbb{E}(Z_1))}{\ln(n)}.$$

Where Z_1 is the number of remaining cuboids after one percolation $Z_1 = |\{(i_1, \dots, i_D) \mid \epsilon_{i_1, \dots, i_D}^1 = 1\}|$. So $\mathbb{E}(Z_1) = pn^D$, and finally:

$$\mathbb{E}(\dim_H(P)) = \frac{\ln(pn^D)}{\ln(n)} = D + \log_n(p).$$

Formally The above argument can be made rigorous. Defining a contractive map that reflects the self similarities; Then use Stefan Banach's contractive mapping fixed point theorem applied to the complete metric space of non-empty compact subsets of R^n with the Hausdorff distance.

However, we choose to show the fractional dimension is $\alpha = D + \log_n(p)$ directly (this proof follows some ideas from [Falconer(1990), p. 34-35, ex. 2.7]).

From definition, $P = \bigcap_{d \rightarrow \infty} P^k$. P^k is composed if cuboids of side $1/n^k$. Let Z^k be the number of these cuboids. In expectation, P^k should have $\mathbb{E}(Z^k) = (pn^D)^k$ cuboids of side $1/n^k$ remaining. Taking P^k as a \sqrt{D}/n^k -cover (let $\beta_k = \sqrt{D}/n^k$ ¹⁰) of P gives

$$\mathbb{E}(H_{\beta_k}^\alpha(P)) \leq (pn^D)^k (\sqrt{D}/n^k)^{D+\log_n(p)} \leq \sqrt{D}^\alpha.$$

Letting $k \rightarrow \infty$, get

$$\mathbb{E}(H^\alpha(P)) \leq \sqrt{D}^\alpha < \infty.$$

¹⁰So that β_k is the diameter of a cuboid of side $1/n^d$ in dimension D .

Let $\mathcal{U} = \{U_i \mid i \in I\}$ be an open cover of P . For each $i \in I$, let $k \in \mathbb{N}$ be such that $\beta_{k+1} \leq \text{diam}(U_i) < \beta_k$. Then, in general, U_i may only cover one cuboid from P^k . As generating P is a random process, it may happen that a U_i covers more than one, but this is not the case in general (as it involves a very particular selection configuration), and will not be consistently the case as we take $k \rightarrow \infty$. Now, if $j \geq k$, then U_i intersects in expectation at most $(pn^D)^{j-k} \leq \frac{(pn^D)^j}{\sqrt{D}^\alpha} \text{diam}(U_i)^\alpha$ cuboids of side $1/n^j$. Taking j such that $\beta_{j+1} \leq \text{diam}(U_i) \quad i \in I$, since \mathcal{U} intersect in expectation $\mathbb{E}(Z^j) = (pn^D)^j$ cuboids of diameter $1/n^j$, we get

$$(pn^D)^j \leq \sum_{i \in I} \frac{(pn^D)^j}{\sqrt{D}^\alpha} \text{diam}(U_i)^\alpha.$$

This implies that in expectation, $\sqrt{D}^\alpha \leq H^\alpha(P)$. Thus, $\mathbb{E}(H^\alpha(P)) \geq \sqrt{D}^\alpha > 0$.

Finally, as $0 < H^\alpha(P) < \infty$, get $\mathbb{E}(\dim_H(P)) = \alpha = D + \log_n(p)$.

2.6 Blob

To have a better understanding, we begin by studying the central "blob" of the fractal.

Definition We define the blob of a percolation P as the connected component of P that contains the point at the centre of the cuboid (that is, the component that contains $(1/2, \dots, 1/2)$). Note that the blob may be empty.

In the simulations, since infinite percolation may only be approximated, we only consider cases where n is odd, so that the cuboid containing the point $(1/2, \dots, 1/2)$ is uniquely well defined.

Algorithm The algorithm we use to find the blob is the following:

2.6.1 Manhattan (step) distance

2.6.2 Euclidean distance

2.6.3 Area

2.6.4 Volume

2.7 Random Walks

3 Percolation Crossings

We are now interested in the existence of crossings from one side of the unit cuboid to the other side.

3.1 Types of Crossings

There will be three types of crossing considered, according how the path of is allowed to behave.

Consider a percolation P on the unit cuboid $[0, 1]^D$ in D dimensions, and $\gamma : [0, 1] \rightarrow [0, 1]^D$ to be a crossing of the cuboid, i.e. $\gamma(0) \in 0 \times [0, 1]^{D-1}$ and $\gamma(1) \in 1 \times [0, 1]^{D-1}$.

In the case of a finite percolation¹¹, we can view a crossing as a walk on the cuboids of side $1/n^d$.

Straight The crossing is said to be straight if $\gamma(x) = x \times (a_2, \dots, a_D)$. That is, if the path goes on a direct line from one side to the other. For finite percolations, this means only one step direction is allowed.

Semi-Straight The crossing is said to be semi-straight if $\forall 0 \leq x < y \leq 1$, $\gamma(x)_0 < \gamma(y)_0$. That is, if the path never goes backwards. For finite percolations, this allows steps in $2D - 1$ direction: all the possible directions except $(-1, 0, \dots, 0)$.

Non-Straight If the crossing is neither straight or semi-straight, then it is non-straight.

The following illustrate the different types of crossings in two dimensions (the easiest to picture), extension to greater dimensions is straightforward.

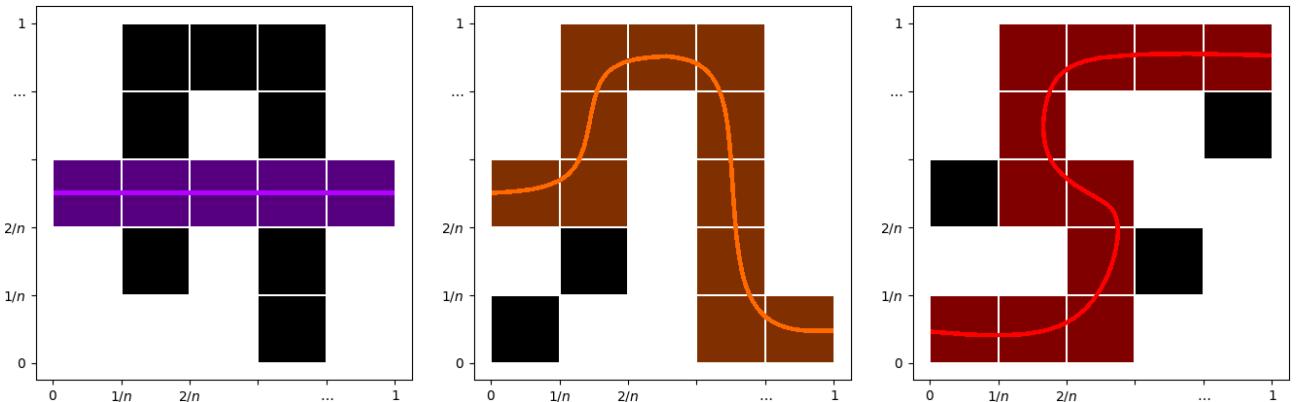


Figure 11: Straight, Semi-Straight, and Non-Straight Crossings

3.2 Finding Crossings

Percolations are complicated objects. Even for finite percolations, say one with side n , depth d in D dimensions, here are $2^{(n^D)^d}$ possibilities. This grows extremely rapidly, and makes theoretical results hard to derive.

Therefore, most of the time, we will use simulations to study these objects.

Examples To see how complicated finding crossing on percolation becomes when d grows, we look at some examples in two dimensions ($D = 2$), with sides $n = 2, 3, 5$ (see fig. 12, 13, 14).

Algorithms

¹¹i.e. $P \sim \text{Perc}^D(n, p, d)$ with $n, d < \infty$

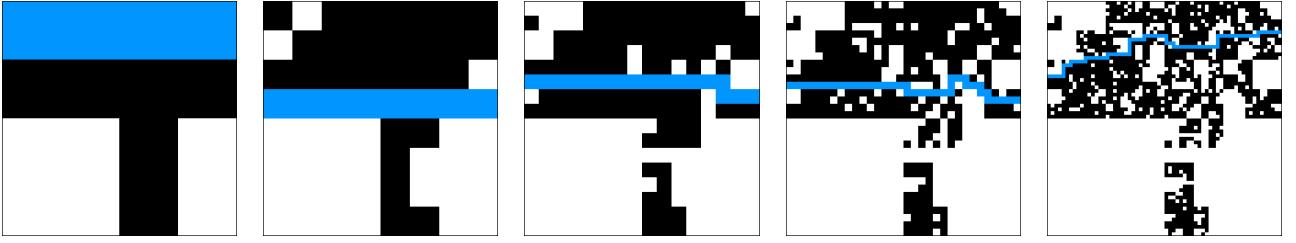


Figure 12: Crossings for a percolation with side $n = 2$, at depths $k = 2, 3, 4, 5, 6$.

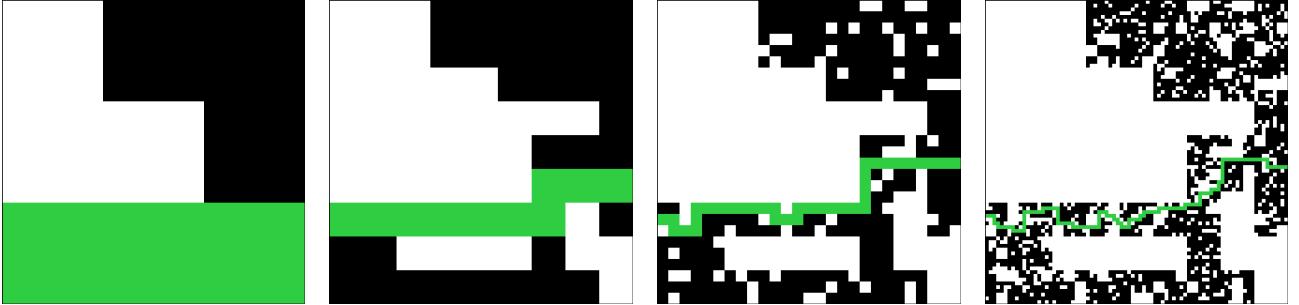


Figure 13: Crossings for a percolation with side $n = 3$, at depths $k = 1, 2, 3, 4$.



Figure 14: Crossings for a percolation with side $n = 5$, at depths $k = 1, 2, 3$.

Percolations As (recursive) percolations are generated through a recursive process, it is logical (and elegant) to use a recursive function to generate them computationally. We will represent a (finite) percolation $P \sim \text{Perc}^D(n, p, d)$ with a binary D -dimensional array of length n^d in each dimensions. Note that D is a super-parameter.

This algorithm is given in pseudocode in algorithm 1. Julia implementations^{12 13} (used later in simulations for empirical estimations) can be found on the GitHub repository¹⁴.

Crossings Given a percolation, it is not clear how to proceed on the task of determining if a crossing exists.

Our approach of the problem is rather unusual.

We simulate a "fire" propagating from one side of the cuboid, and check if it reaches the other

¹²2D percolation Julia implementation: https://github.com/pauldubois98/PercolationFractalsStudy/blob/main/fractal_percolation2D.jl

¹³3D percolation Julia implementation: https://github.com/pauldubois98/PercolationFractalsStudy/blob/main/fractal_percolation3D.jl

¹⁴See <https://github.com/pauldubois98/PercolationFractalsStudy>.

¹⁵`random()` generates a random number $x \sim \mathcal{U}(0, 1)$ ¹⁶.

¹⁶Writing $x \sim \mathcal{U}(a, b)$ for x a random variable with uniform distribution on $[a, b]$.

Algorithm 1 Percolation generation algorithm

```

1: Fills the  $[i_1, i_1 + n^k] \times \dots \times [i_D, i_D + n^k]$  sub-array of  $P$ .
2: procedure PERCOLATIONFILLD( $P$ ,  $i_1, \dots, i_D$ ,  $n, p, k$ )
3:   if  $k = 0$  then
4:      $P[i_1, \dots, i_D] \leftarrow \text{random}()^{15} < p$                                  $\triangleright$  stop recursion
5:   else
6:     for  $0 \leq j_1, \dots, j_D < n$  do
7:       if  $\text{random}() < p$  then
8:         PERCOLATIONFILL( $P$ ,  $i_1 + (j_1 n^k), \dots, i_D + (j_D n^k)$ ,  $n, p, k - 1$ )       $\triangleright$  recursion
9:       else
10:         $P[i_1 : i_1 + n^k, \dots, i_D : i_D + n^k] \leftarrow \text{zeros}^D(n^k, \dots, n^k)$        $\triangleright$  void cells
11:       end if
12:     end for
13:   end if
14: end procedure
15: Generates a percolation  $P \sim \text{Perc}^D(n, p, d)$ .
16: procedure PERCOLATIOND( $n, p, d$ )
17:    $P \leftarrow \text{init}^D(n^d, \dots, n^d)$ 
18:   PERCOLATIONFILL( $P$ ,  $i_1, \dots, i_D$ ,  $n, p, d$ )                                 $\triangleright$  fill  $P$  recursively
19:   return  $P$ 
20: end procedure

```

side.¹⁷ If it does, then a crossing exists, otherwise, no crossing exists. The rules of propagation for the fire (materialized by active cells¹⁸) are as follows:

1. If a cell is active at time t , then the non-empty adjacent cells become active at time $t + 1$.
2. If a cell is active at time t , it deactivates and can no longer be active on times $\tilde{t} > t$.

At the beginning ($t = 0$), we set all the non-empty cells from one side of the unit cuboid as active cells. As soon as a cell from the opposite side of the unit cuboid is active, a path has been found (the time t even gives the length of the crossing), and this is the shortest crossing. If at some time t , no cell is active and no crossing was found before (i.e. the blaze turns off before reaching the other side), then no crossing exist.

Note that this technique tells us if there is a crossing, and what is the minimal length. However, it does not give the crossing path explicitly. To find it, one needs to save the set of active cells at each time, and propagate backwards.

This algorithm 2 is a pseudocode implementation of this process. Julia implementations^{19 20} (used for empirical probability approximation) can be found again on the GitHub repository²¹.

Visualisation of this algorithm:

- In 2D: <https://pauldubois98.github.io/PercolationFractalsAlgorithmsDemo/2Dcrossing/index.html>

¹⁷This is inspired by simulations of forest fires from the game "The Pyromaniac Game" [pyr(2017)].

¹⁸We call a "cell" one of the D dimensional cuboids of side $1/n^d$

¹⁹2D crossing Julia implementation: <https://github.com/pauldubois98/PercolationFractalsStudy/blob/main/crossings2D.jl>

²⁰3D crossing Julia implementation: <https://github.com/pauldubois98/PercolationFractalsStudy/blob/main/crossings3D.jl>

²¹See <https://github.com/pauldubois98/PercolationFractalsStudy>.

- In 3D: <https://pauldubois98.github.io/PercolationFractalsAlgorithmsDemo/3Dcrossing/index.html>

Algorithm 2 Crossing finding algorithm

```

1: Perform one step propagation on A, with respect to P
2: procedure NEIGHBORSD(A, P, n, d)
3:   B  $\leftarrow$  zerosD(nd, ..., nd) ▷ newly active cells
4:   for 1  $\leq$  i1, ..., iD < nd do
5:     if A[i1, ..., iD] = 2 then
6:       B[i1, ..., iD] = 2 ▷ remain inactive
7:     end if
8:     if A[i1, ..., iD] = 1 then
9:       B[i1, ..., iD] = 2 ▷ deactivate cell
10:      for 1  $\leq$  j  $\leq$  D do
11:        if P[i1, ..., ij - 1, ..., iD] and A[i1, ..., ij - 1, ..., iD] = 0 then
12:          B[i1, ..., ij - 1, ..., iD]  $\leftarrow$  1 ▷ activate cell
13:        end if
14:        if P[i1, ..., ij + 1, ..., iD] and A[i1, ..., ij + 1, ..., iD] = 0 then
15:          B[i1, ..., ij + 1, ..., iD]  $\leftarrow$  1 ▷ activate cell
16:        end if
17:      end for
18:    end if
19:  end for
20:  return B
21: end procedure
22: Returns the length of the shortest crossing on percolation P  $\sim$  PercD(n, p, d); returns 0 if none exist.
23: procedure CROSSINGD(P, n, d)
24:   A  $\leftarrow$  zerosD(nd, ..., nd) ▷ active cells
25:   A[0, :, ..., :]  $\leftarrow$  P[0, :, ..., :] ▷ initialize active cells
26:   t  $\leftarrow$  0
27:   while any(A = 1) and any(A[nd, :, ..., :] = 1) do
28:     A  $\leftarrow$  NEIGHBORS(P, A) ▷ propagate one step
29:     t  $\leftarrow$  t + 1
30:   end while
31:   if any(A[nd, :, ..., :]) then
32:     return t
33:   else
34:     return 0
35:   end if
36: end procedure

```

By restricting the possible steps (not allowing activation to propagate backwards), this algorithm may be adapted to find semi-straight crossings (the above would give non-straight crossings). It can also be used for straight crossings, but in this case, it is faster to sum the percolation array on the first dimension: a straight crossing exists if one of the sums gives n^d .

3.3 Crossings Probability

For a percolation $P \sim \text{Perc}^D(n, p, d)$, we look at the probability that a crossing exist.

3.4 Crossings Length

3.5 Crossings Dimension

4 Projections

4.1 2D to 1D

4.2 3D to 2D

4.3 3D to 1D

4.4 Dimension of the (non void) Universe

References

- [Ionescu et al.(2009)] Ionescu, Oustaloup, Levron, Melchior, Sabatier, and De Keyser] Clara Ionescu, Alain Oustaloup, François Levron, Pierre Melchior, Jocelyn Sabatier, and Robin De Keyser. A model of the lungs based on fractal geometrical and structural properties. *IFAC Proceedings Volumes*, 42(10):994–999, 2009. ISSN 1474-6670. doi: <https://doi.org/10.3182/20090706-3-FR-2004.00165>. URL <https://www.sciencedirect.com/science/article/pii/S1474667016387791>. 15th IFAC Symposium on System Identification.
- [Pol(2009)] Lectures on fractals and dimension theory, 2009.
- [3DX(2020)] *Mathematical Documentation of objects*, 2020.
- [Shishikura(1992)] Mitsuhiro Shishikura. The Hausdorff dimension of the boundary of the Mandelbrot set and Julia sets. *arXiv Mathematics e-prints*, art. math/9201282, April 1992.
- [Heinemann and Stratmann(1998)] Stefan M. Heinemann and Bernd O. Stratmann. Quadratic polynomials with julia sets of hausdorff dimension close to 2, 1998.
- [Falconer(1990)] Kenneth Falconer. *Fractal Geometry*. Wiley, 1990. ISBN 0978-0-471-92287-2.
- [pyr(2017)] The pyromaniac game, 2017.



UNIVERSITY OF OXFORD
MATHEMATICAL INSTITUTE

Random Fractals
Appendix

Author
Paul DUBOIS

Supervisor
Ben HAMBLY

April 26, 2021

A More on Fractals

Here will be discussed some extensions of famous fractals studied before (in 1.2.2). This is not directly relevant for this paper, but an involved reader may be interested.

A.1 Mandelbrot and Julia Sets

Mandelbrot and Julia sets involve calculating the limit of a sequence to determine if a point is in the set or not. More precisely, one needs to determine whether the sequence diverges or not. It is impossible to do such calculations with a computer, therefore, we approximate. For each initial value z_0 , we calculate the first values of the sequence. We stipulate that the time taken by the sequence to reach $|z_n| > R$ is an indication of how fast the sequence diverges.

The plots are then made as follows: We calculate the first M values of the sequence. If $|z_n| > R$ for some $n < M$, then let $g(z_0) = n$, with n the smallest such integer. If $|z_n| \leq R$ for all $n < M$, then let $g(z_0) = M$. Finally, plot g on the area $S \subseteq \mathbb{C}$ of interest.

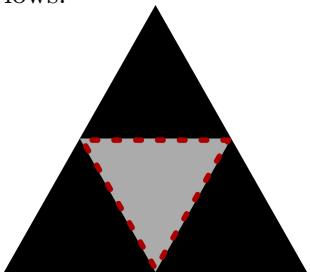
On the figures 1 and 2, the settings are $R = 2$ and $M = 100$. The plot represent the value of g through colours, with the same colour-bar as in fig. 15.

A.2 Sierpiski Fractals

Following the idea of Sierpiski carpet, several fractals may be generated. First, using (equilateral) triangles instead of squares:

Sierpiski Triangle Sierpiski triangle is also a fractal constructed recursively by removing parts of its initial set, following a pattern similar to the one for Sierpiski carpet (although, there are equivalent definitions).

The construction starts from an equilateral triangle. Then, at each step, split every equilateral triangle into 4 sub-triangles as follows:



Remove the central triangle, and repeat the operation on the remaining 3 triangles.

The figure is made of 3 copies of itself, scaled by a factor of $\frac{1}{2}$. Therefore, the intuitive Hausdorff dimension for this set is $\dim_H(K) = \frac{\log(3)}{\log(2)} \approx 1.585$. Again, this can be proved rigorously using similar techniques as in [Falconer(1990), p. 34-35, ex. 2.7].

Sierpiski 3D Fractals Seen previously, Sierpiski carpet and triangle live in a 2D world, but the idea of Sierpiski fractal may also be extended to 3D.

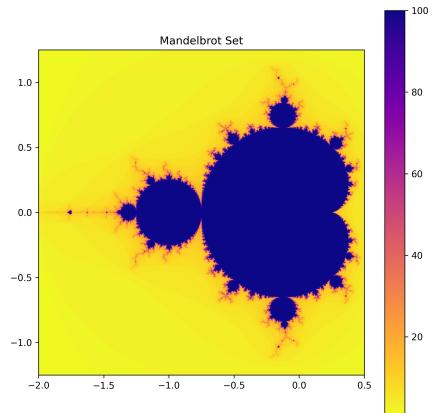


Figure 15: Mandelbrot Set

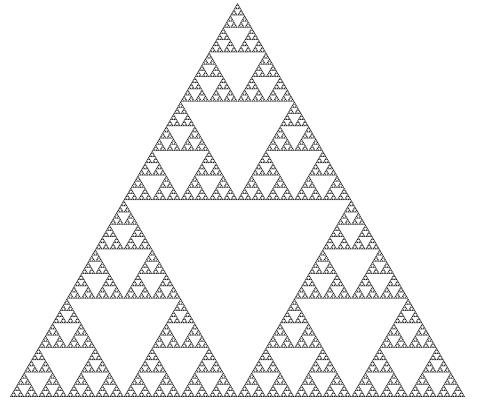


Figure 16: Sierpinski Triangle (8 steps)

Menger sponge The Menger sponge is the generalisation of Sierpiski carpet to 3 dimensions.

Start with a cube; split it into 27 identical copies scaled by $\frac{1}{3}$, and remove the central one, and the 6 cubes sharing a face with the central cube. Repeat this operation on each of the 20 remaining cubes.

From this construction arise a fractal with dimension $\frac{\log(20)}{\log(3)} \approx 2.727$.

Sierpiski tetrahedron The Sierpiski tetrahedron (or tetrix) is the analogue of Sierpiski triangle in 3 dimensions. Start with a regular tetrahedron; split it into 5 identical copies scaled by $\frac{1}{2}$, and remove the central one. Repeat this operation on each of the 4 remaining tetrahedrons.

From this construction arise a fractal with dimension $\frac{\log(4)}{\log(2)} = 2$. Note that this set is dimension exactly 2 while not being a surface.

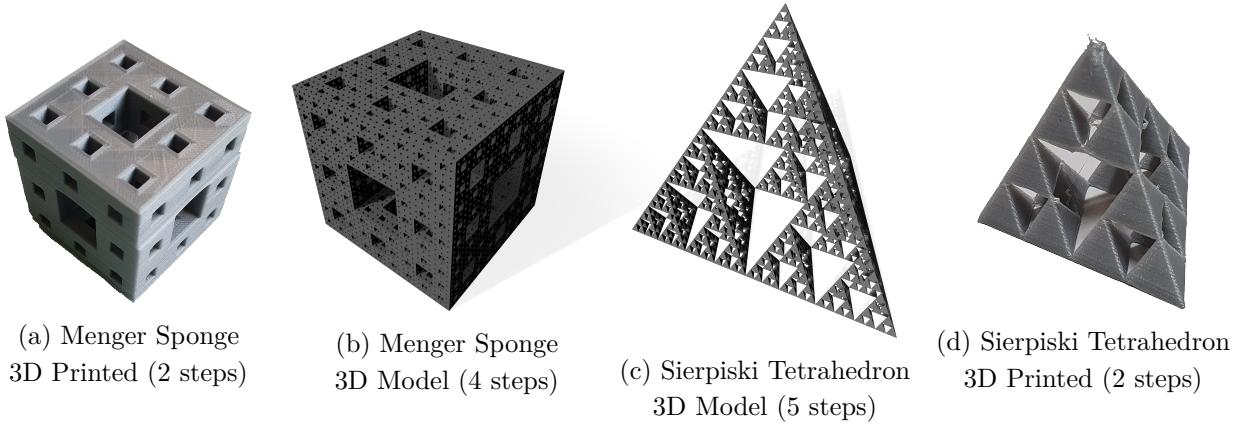


Figure 17: Sierpiski 3D Fractals

A.3 Coastline of Islands

The coastline can be measured at different scales. As the scale is refined, one measures a more detailed curve, which leads to a larger length. This refinement can be done indefinitely: from segments of 1000km (one would only capture the rough coastline shape) to segments of 1mm (one would need to go around every grain of sand), and even further considering sub-atomic scales.

Following this reasoning, the (1D) length of coastlines is infinite. It is in fact a fractal with dimension greater than one (but less than 2, as contain in a plane).

It is then possible to measure how disordered the coastline of an island is, by approximating its Hausdorff dimension. This will be done macroscopically, using maps extracted from the Google Maps service. In this paper, 3 islands will be considered: The United Kingdom, Iceland, and Madagascar. The United Kingdom and Iceland are understood as having messy coastlines (so should have a high Hausdorff dimension), whereas Madagascar is thought to have a rather smooth coastline (Hausdorff dimension close to one).

Of course, fractal dimension can only be an approximation. We approach the problem as follows: First, we take a high resolution map of the island, and split it into a binary colour image, black for the ground, and white for the surrounding sea (as in fig. 18). Then, we reduce the size of the image by merging pixels ("pixelizing") to obtain lower resolution images. The resolution panel will correspond to the measuring scales panel. A subset of the resulting images is shown in fig. 19.

Now, using computer vision (as a Python package), we obtain the contours of the shape in an image. It is then straightforward to measure the length of the contour (in pixels, which can be

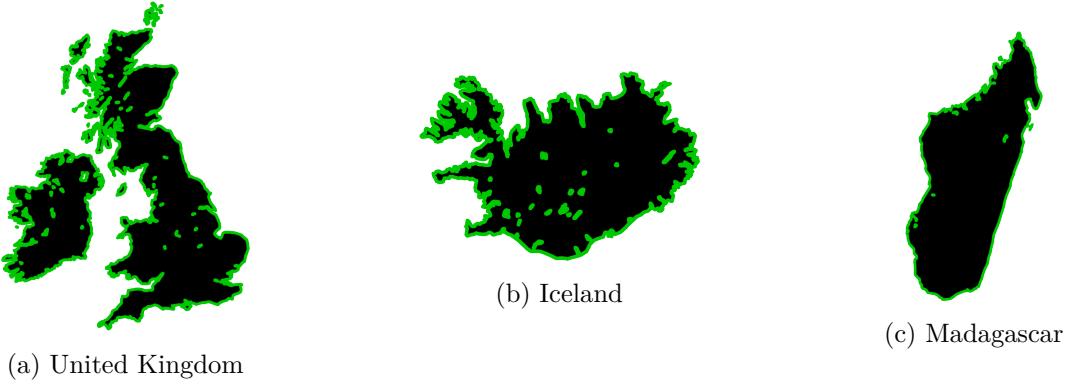


Figure 18: Coastlines (coloured in green).



Figure 19: Map scales for the UK, Iceland, and Madagascar.

converted to kilometres as we know the scale of the image). Repeating this operation for each scale of the map gives an array of coastline length with associated scales.

Plotting the values in a log-log plot show a nearly straight line; the approximate dimension will be given by the slope of the best fitting line. We obtained the following dimensions:

United Kingdom $D \approx 1.2354$

Iceland $D \approx 1.2466$

Madagascar $D \approx 1.0600$

The fitting line is very close to the data (see fig. 20), showing that the behaviour is as expected.

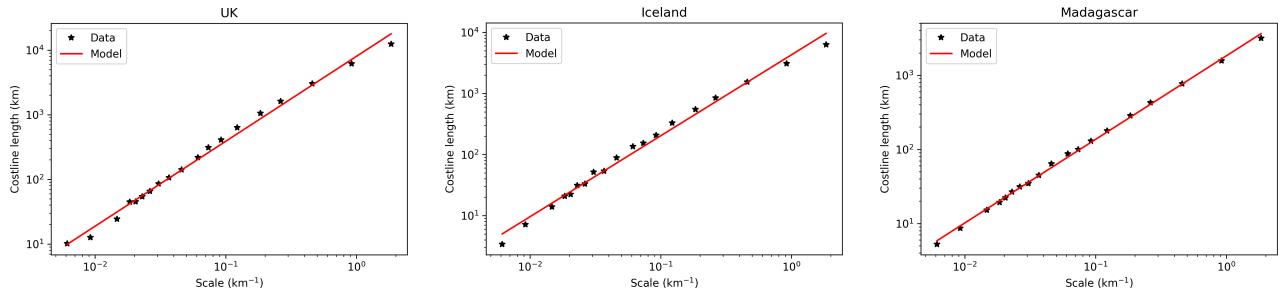


Figure 20: Islands Dimension Regression

The full code for this study can be found here: <https://colab.research.google.com/drive/1qo8S8oxqcLsw9UFrq5wTuVyxVA48uwFF?usp=sharing>.

B Plots

C Codes